***Nivel de Enlace***

Alumno: Manuel Cantonero Chamorro / Mario Corchero Jiménez

DNI: 09204648 W

Asignatura: **Interconexión de sistemas**

Curso: **2009/2010**

Grupo de prácticas: Lunes

Profesor: Manuel Díaz Díaz

Fecha: 1 de Junio de 2010

**INDICE**

[1. Manual de usuario 2](#_Toc262687195)

[1.1. Introducción 2](#_Toc262687196)

[1.2. Guía de Instalación 2](#_Toc262687197)

[1.3. Interfaz de usuario 2](#_Toc262687198)

[1.3.1. Interfaz Principal 2](#_Toc262687199)

[1.3.2. Interfaz de Configuración 3](#_Toc262687200)

[1.3.3. Interfaz de conexión 3](#_Toc262687201)

[1.3.4. Petición de conexión 4](#_Toc262687202)

[1.3.5. Petición de desconexión 4](#_Toc262687203)

[1.4. Ejemplo de funcionamiento 4](#_Toc262687204)

[1.5. Lista de errores 4](#_Toc262687205)

[1.6. Historial de desarrollo 4](#_Toc262687206)

[1.7. Valoración Final del proyecto 4](#_Toc262687207)

# Manual de usuario

## Introducción

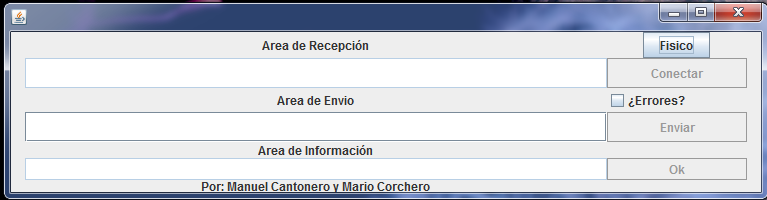
Se ha desarrollado la implementación de un protocolo de comunicación entre dos ETD’s cualquiera en configuración punto a punto o multipunto de estaciones combinadas en enlace directo a través del “cable módem nulo”, para trasmisión semiduplex con paro y espera y envío bidireccional de información con errores y corrección por retransmisión y resolución de colisiones por vencimiento de plazos y numero de intentos. Para la implementación de el nivel de enlace, ha sido necesario también la implementación de un nivel físico, utilizando la librería RXTX para la comunicación entre puertos serie. También ha sido necesario desarrollar una interfaz que hará del nivel superior (controlada por el usuario). La aplicación se ha desarrollado en Java.

## Guía de Instalación

La instalación de la aplicación consiste en la copia de los binarios y la ejecución de la clase entorno en la maquina virtual java. Es necesario tener instalada en el ordenador la librería RXTX.

## Interfaz de usuario

### Interfaz Principal



La interfaz se compone de la siguientes pares.

Áreas:

-Área de Recepción: aparecerá la información relacionada a la recepción de tramas y el control del temporizador y el número de intentos. Se mostraran las tramas de datos, y la interpretación de las tramas de control. (Trama anterior confirmada, trama de fin de comunicación finalizada, etc.) [Solo lectura]

-Área de Envío: zona para escribir los datos que se desean enviar al otro ETD. [Solo escritura]

-Área de información: aparecerá otra información relevante. La trama construida que se va a enviar, si se ha realizado la conexión correctamente, etc.

[Solo lectura]

Botones:

-Físico: Botón solo activo al principio, permite configurar el nivel físico. Es necesario pulsar este botón para que se active el botón de conectar o para poder recibir una petición de conexión.

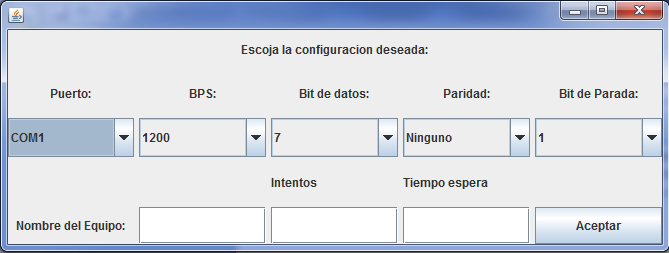
-Conectar: Una vez conectados al físico, nos permitirá conectarnos a otro ETD para enviar datos. Una vez conectados, Este botón se transformara en el botón Finalizar, para cortar la comunicación con el otro ETD. (No está siempre disponible)

-Enviar: Botón para generar la trama y que aparezca en el área de información (lista para ser enviada), solo se activa cuando es nuestro turno.

-OK: Botón para confirmar el envío de la trama mostrada en el área de información. (Solo activo cuando hay una trama lista para enviar)

-Errores: Permite activar el envío de tramas de datos con errores, cuando esta activado, las tramas que se envíen contendrán errores (una @ en una posición aleatoria de los datos).

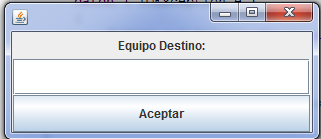
### Interfaz de Configuración



Se abre al pulsar el botón Físico.

Permite configurar valores del físico (parte superior) y del nivel de enlace (parte inferior). Si no se introduce un valor en intentos o en tiempo de espera, se tomaran valores por defectos: 15 segundos y 3 intentos. En nombre del equipo deberemos de insertar un carácter, en caso de meter más de un carácter, el programa informara del error.

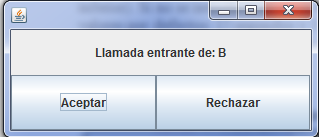
### Interfaz de conexión



Se abre al pulsar el botón de Conectar.

Debemos introducir el nombre del ETD destino. Si introducimos más de un carácter, el programa nos informará del error.

### Petición de conexión

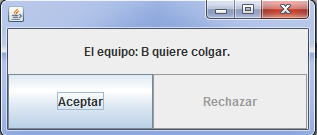


Se abre al recibir una trama de conexión.

Podemos pulsar aceptar, para enviar la aceptación o rechazar para enviar el rechazo.

Arriba nos aparecerá el nombre del equipo que nos llama.

### Petición de desconexión



Se abre al recibir una trama de fin de comunicación.

Solo podremos aceptar.

## Lista de errores

-Si el puerto al que nos conectamos está ocupado, el programa informará del problema y no se podrá conectar al físico.

-Introducir el nombre de un ETD con más de un carácter. El programa informara del error y no permitirá conectar.

-No se tiene la librería RXTX instalada. El programa informara del error en el driver y no podrá iniciar la aplicación.

## Características a destacar

Este apartado vamos a resaltar algunas peculiaridades que puede tener nuestra práctica frente a otras.

* Nuestra práctica la hemos realizado en Java, usando el entorno de programación Eclipse. Un aspecto muy importante a destacar es la utilización de una librería específica para poder conectarnos con el puerto serie, esta librería es la RXTX.  
  Hemos utilizado Java, ya que el menú mostrado para la interacción y la visualización de la práctica, es mucho más claro, sencillo, breve, y desde nuestro punto de vista es más fácil de entender, haciendo de esta manera que el usuario pueda interactuar con mayor facilidad y entienda rápidamente la práctica.
* Como hemos comentado anteriormente hemos utilizado la librería RXTX para poder abrir los puertos e interactuar con ellos. Es necesario tener esta librería instalada y bien configurada para poder tener un uso correcto de nuestra práctica.
* Otra característica por la que hemos usado Java, es la facilidad con la que podemos tratar las cadenas, ahorrándonos mucho tiempo y líneas de código en estas implementaciones.
* Es importante comentar que en la opción de introducir los errores, éstos se introducen en una posición aleatoria, de esta forma hemos incluido una mejora (creemos nosotros) en la implementación de nuestra práctica. Ya que puede ocurrir que alguien solo los introduzca en una misma posición, realizando los cálculos solo con ella y de esta manera no entender correctamente el trabajo realizado sobre la detección de errores.
* Otra característica de usar Java es la posibilidad de poder lanzar procesos concurrentemente, un ejemplo sería la utilización del temporizador (controlado a través de eventos), pudiéndonos olvidar de él, hasta que finalice el tiempo o realicemos otra opción.
* En un principio, nuestra práctica no hacía exactamente lo establecido por el profesor, dejando una pequeña “libertad” a la hora de enviar tramas, de esta forma, cualquier usuario podía mandar Datos una vez establecida la conexión. Cambiamos lo realizado por nosotros y la pusimos como se indica en la documentación.
* Como último dato sobre la implementación en java, remarcar que nuestra practica es portable a cualquier sistema.
* Podemos terminar diciendo que la práctica esta implementada siguiendo las especificaciones indicadas por el profesor, funcionando (en nuestra opinión) correctamente.

## Crítica de la práctica

La práctica ha sido interesante, aunque desde nuestro punto de vista, debería de estar un poquito más organizada en la explicación. Es decir poder tener unos puntos principales y bien ordenados, de esta forma no habría mal entendimientos de cómo realizar dicha práctica.

Desde un punto de vista positivo, podemos decir, que esta práctica nos ha servido para entender la implementación de un protocolo, que nos vendrá muy bien para poder entender la parte teórica de la asignatura.

En un punto negativo, debemos decir que al realizarla en Java, nos ha llevado mucho tiempo poder configurar los ordenadores para el correcto funcionamiento de nuestra práctica. Desde nuestro punto de vista todas las librerías deberían de estar instaladas y bien configuradas para poder realizar el proyecto en otro lenguaje como por ejemplo Java.

Para finalizar debemos destacar que nos ha parecido una práctica interesante, con una dificultad no muy elevada, pero que se ha hecho un poquito pesada a la hora de tener que configurar los ordenadores y tener que buscar una librería que funcionara correctamente, todo esto mejoraría si se pudiera contar con una librería para Java desde un principio.